

附录 J: Eclipse 教程

By Y.Daniel Liang

付蓉 译

该帮助文档包括以下内容:

- Eclipse 入门
- 选择透视图
- 创建项目
- 创建 Java 程序
- 编译和运行 Java 程序
- 从命令行运行 Java Application
- 在 Eclipse 中调试

» 提示: 在学习完第一章后使用本教程第 1 节~第 6 节, 学习完第二章后可配合本教程的第 7 节, 开始学习第十四章时可配合本教程的第 8 节。

0 简介

该教程的使用者包括正在通过使用 Eclipse 来学习 Java 课程的学生和想要用 Eclipse 开发 Java 项目的程序员。Eclipse 是由 IBM 提供的一个开源的 Java 程序开发软件。

你也可以使用 JDK 命令行工具来写 Java 程序。JDK 命令行工具包括一系列独立的程序, 如编译器和解释器, 都可从命令行来激活。除了 JDK 命令行工具, 市场上还有很多 Java 的开发工具, 包括 Borland 公司的 JBuilder、NetBeans、Sun 公司的 ONE Studio (商业版的 NetBeans)、Eclipse 和 WebGain Visual Café。这些工具为快速开发 Java 程序提供集成的开发环境 (IDE)。编辑、编译、构造、调试和在线帮助被集成在一个用户图形界面。有效的使用将会极大的提高编程速度。

这个简短的教程将会帮助你熟悉 Eclipse, 尤其是在创建项目、创建程序、编译和运行程序方面。

提示: Eclipse 可运行在任何的 Java 虚拟机之上。本文中所有的屏幕截图都来自于运行在 Windows 之上的 Eclipse3.0。你可以到 www.eclipse.com 下载 Eclipse。

安装提示：安装 Eclipse 之前必须安装 JDK1.5。JDK1.5 可以从 <http://java.sun.com/j2se/1.5/download.html> 下载。Windows 版本的 Eclipse 存放在压缩文件 eclipse-SDK-3.0-win32.zip 中。将文件解压缩到 c:\ 下，解压缩后所有的文件都放在 c:\eclipse 中。

1 开始使用 Eclipse

假设你已经将 Eclipse 安装在目录 c:\eclipse 下。要启动 Eclipse，双击 c:\eclipse 目录下的 eclipse 图标（如图 1）：

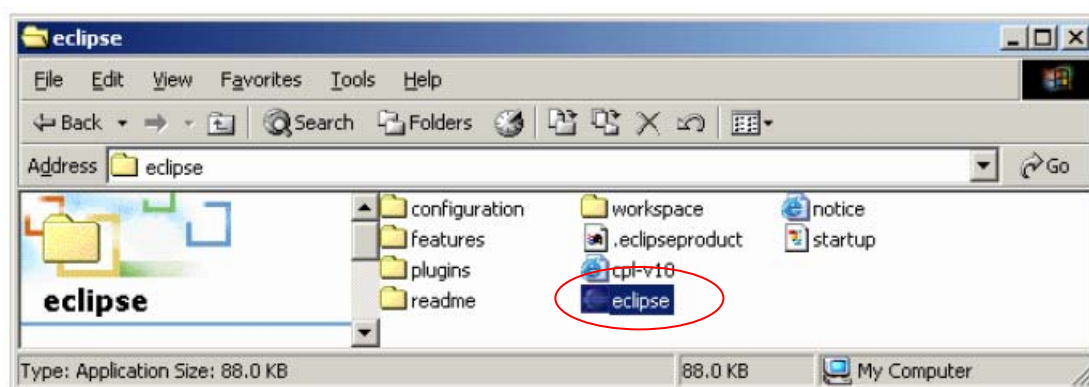


图 1

双击后出现了工作区装载窗口（如图 2）：

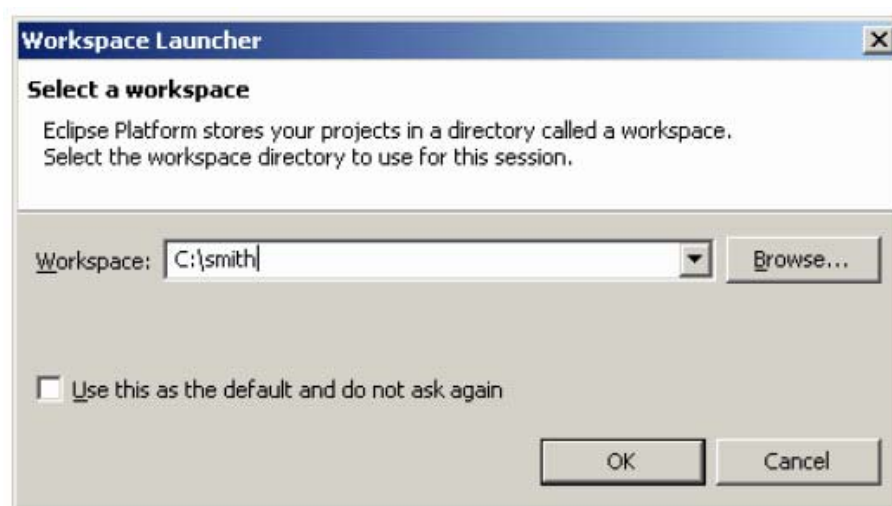


图 2

输入你的工作区所在的目录，本例所有项目、程序都存放在 c:\smith 目录下，故输入 c:\smith，然后点击 OK，Eclipse 的图形界面就展现在你的眼前了（如图 3）。

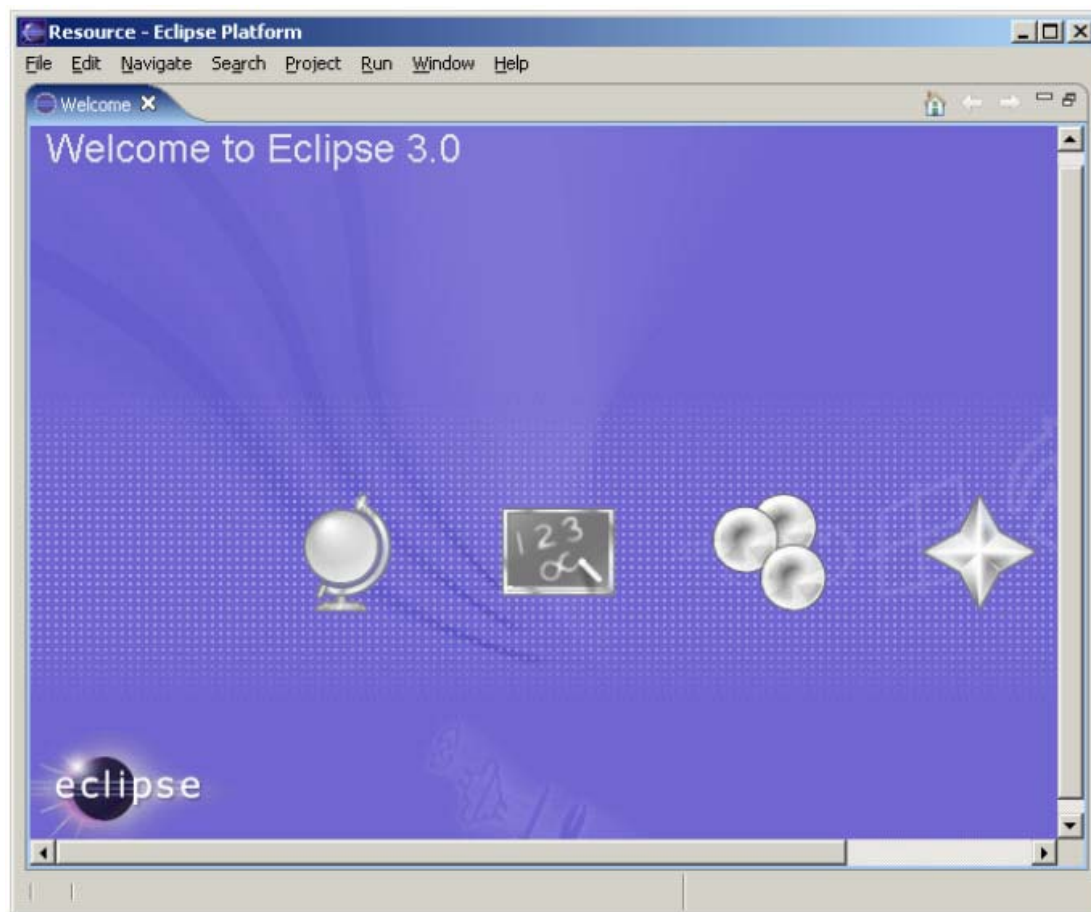


图 3

如果你的工作区已经有项目存在，则项目就会显示在图形界面上。工作区实际上是一个存放项目文件的目录。

2 选择一种透视图

透视图是为了定义 Eclipse 在窗口里显示的最初的设计和布局。透视图主要控制在菜单和工具上显示什么内容。比如，一个 Java 透视图包括常用的编辑 Java 源程序的视图，而用于调试的透视图则包括调试 Java 程序时要用到的视图。你可以转换透视图，但是必须为一个工作区设置好初始的透视图。

要创建 Java 程序，需要从主菜单选择 *Window*，然后选择菜单项 *Open Perspective*，最后选择 *Java*（如图 4）。

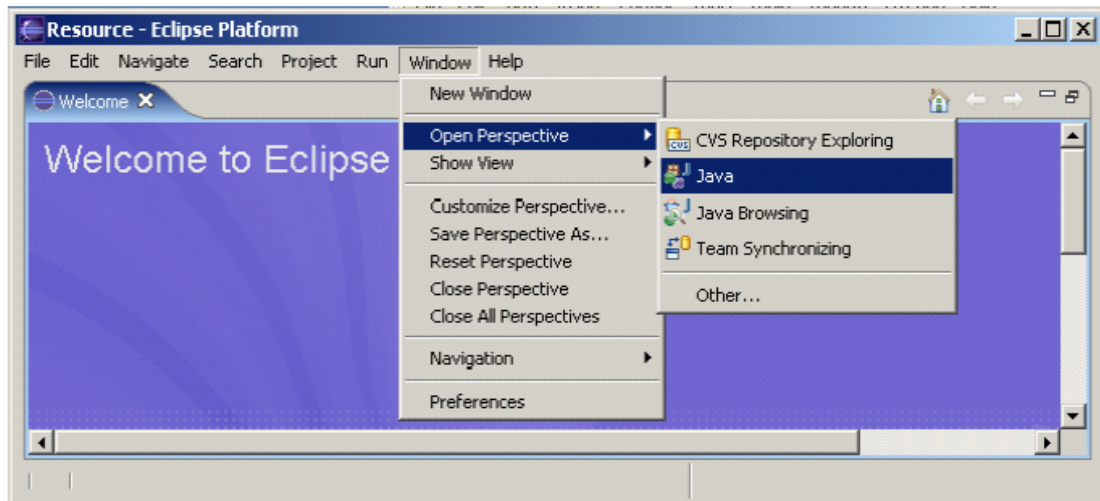


图 4

设置完毕后我们看到用户界面如图 5:

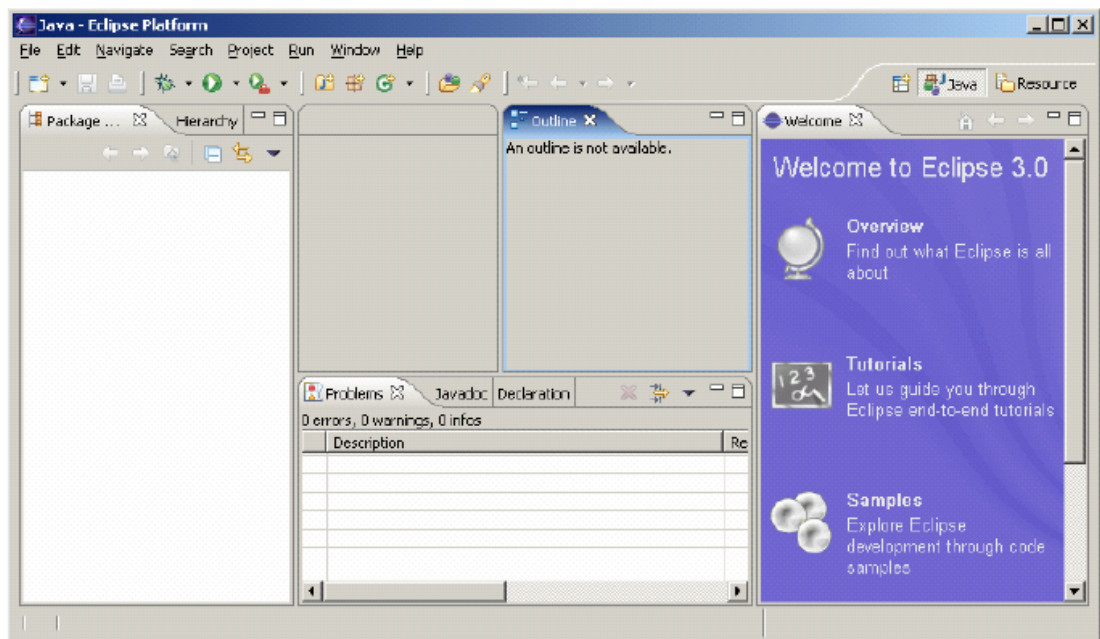


图 5

3 创建一个项目

要创建一个新项目，需要选择菜单项 *File -> New -> Project* 来启动新项目创建向导，如图 6 所示。

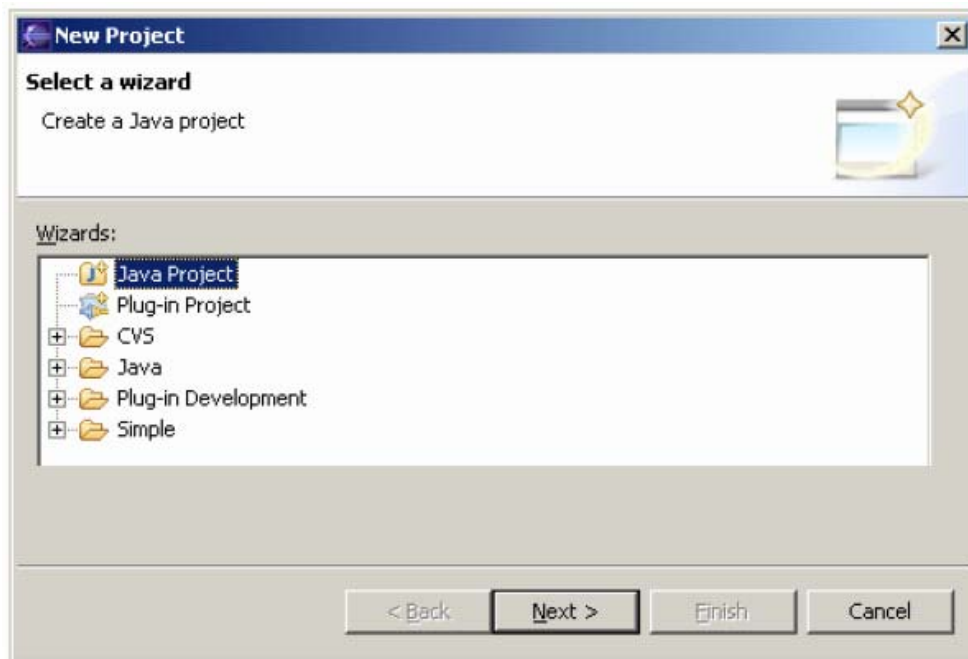


图 6

选择 **Java Project** 然后点击按钮 **Next** 进入下一步，见图 7：

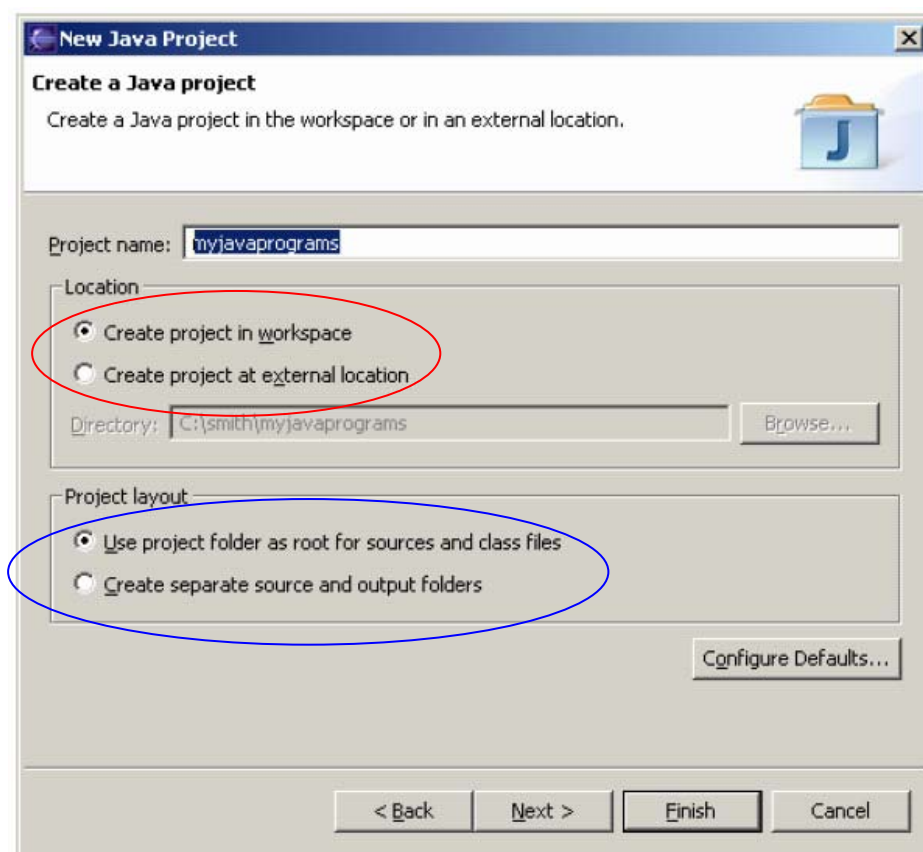


图 7

在 Project name 处键入项目名称 **myjavaprograms**，在你键入的同时 Directory 处的路径也自动更新为 **C:\smith\myjavaprograms**，确定你选择了图 7 红色圈中的选项 **Create**

project in workspace 和蓝色圈中的选项 Use project folder as root for sources and class files。点击按钮 Finish 完成项目的创建。

4 创建程序

项目创建完毕后就可以在这个项目中创建程序了，选择菜单项 *File -> New -> Class* 来启动 Java 类创建向导，如图 8。

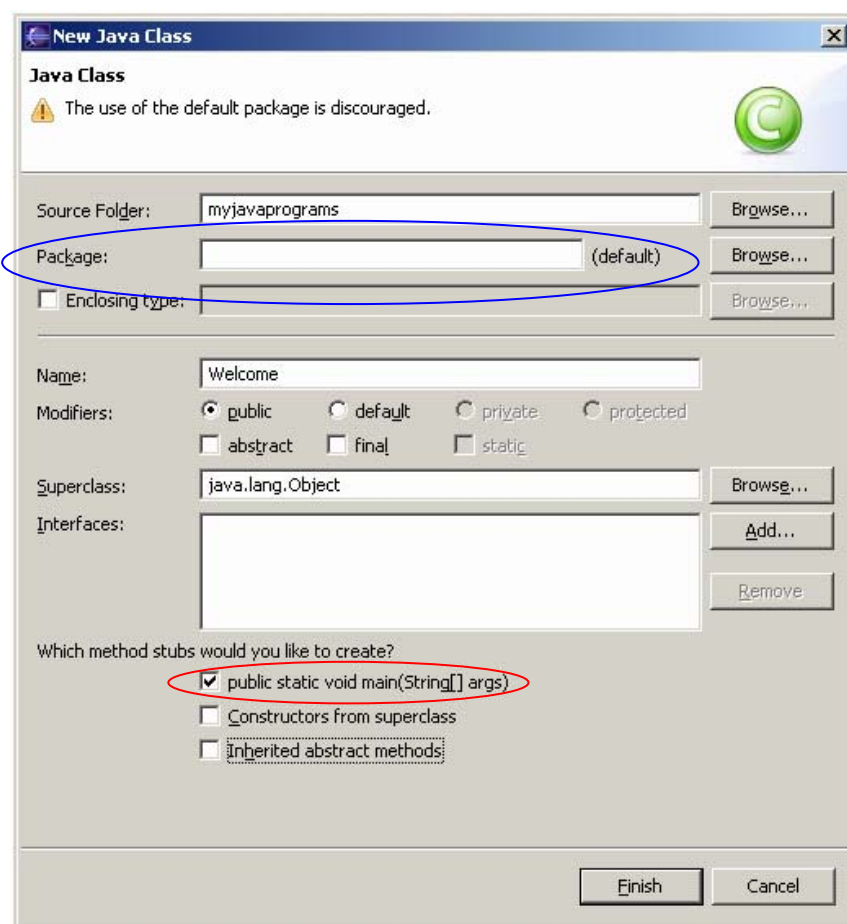


图 8

在 Name 后面的文本框中输入 Welcome，检查是否选中了选项 public void main(String[] args)。点击按钮 Finish 从 Java Application 的模板创建一个文本 Welcome.java，如图 9。

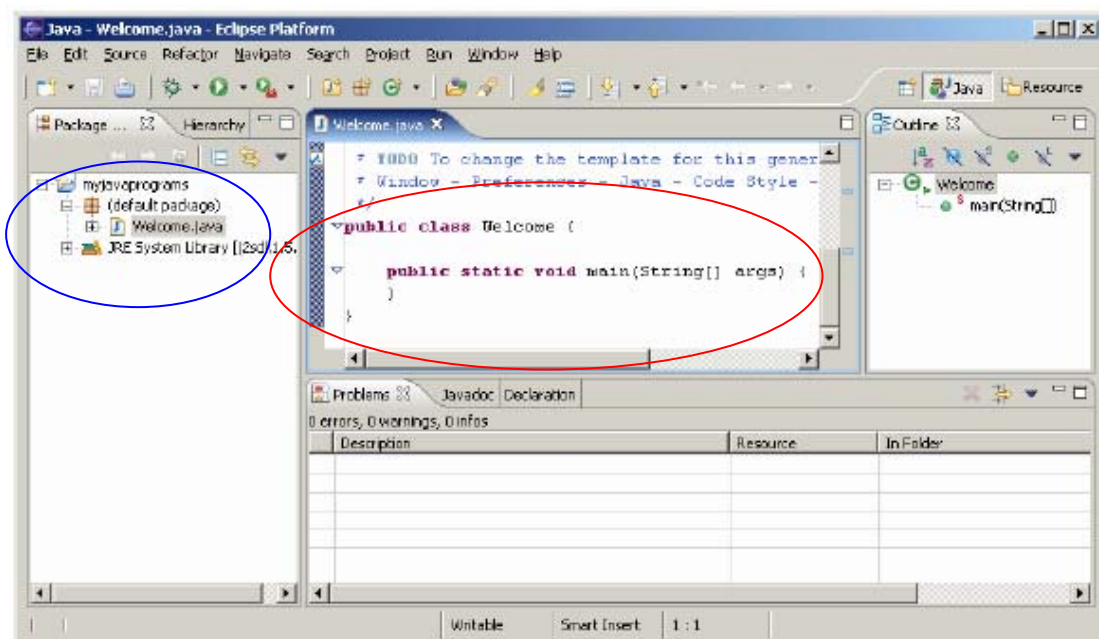


图 9

提示：你可以在图 8 中 Package 后的文本框中输入包名来创建一个包放置 Java 程序（见图 8 蓝色圈）。因为本书中的 Java 源程序没有用到包，所以 Package 后的文本框没有填入内容。对应到图 9，由于用户没有输入 Welcome.java 所属的包，所以系统将这类程序都放到系统的默认包（default package）中，见图 9 蓝色圈中内容。

在程序 Welcome.java 的 main 方法中输入语句 **System.out.println("Welcome to Java");**。

提示：在你键入程序的同时，代码帮助工具会自动给你提示来帮助你完成程序的编写。比如当你在 System 后面键入点（.）后稍等 1 秒，Eclipse 就会显示一个弹出菜单来帮你完成代码，如图 10 所示。你可以从 Eclipse 提供的下拉菜单中选择合适的选项来完成代码。

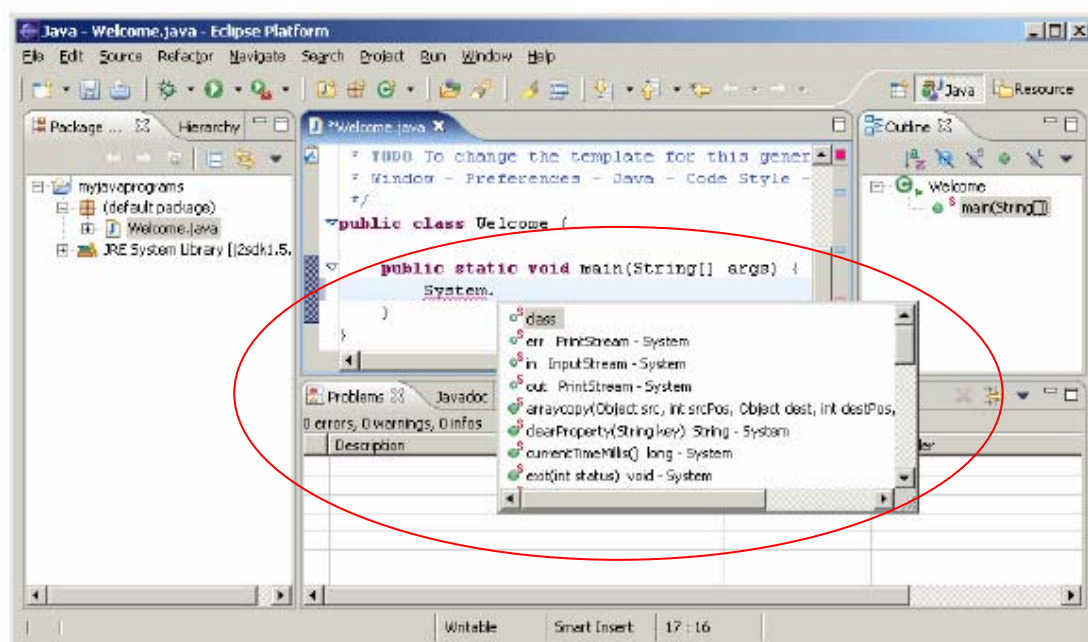


图 10

5 编译和运行程序

一般情况下，源代码在你键入的时候就会自动进行编译。比如，如果你忘记在语句结束的时候键入分号 (;)，见图 11，那么你就会看到一条红色的波浪线来指出你的错误。

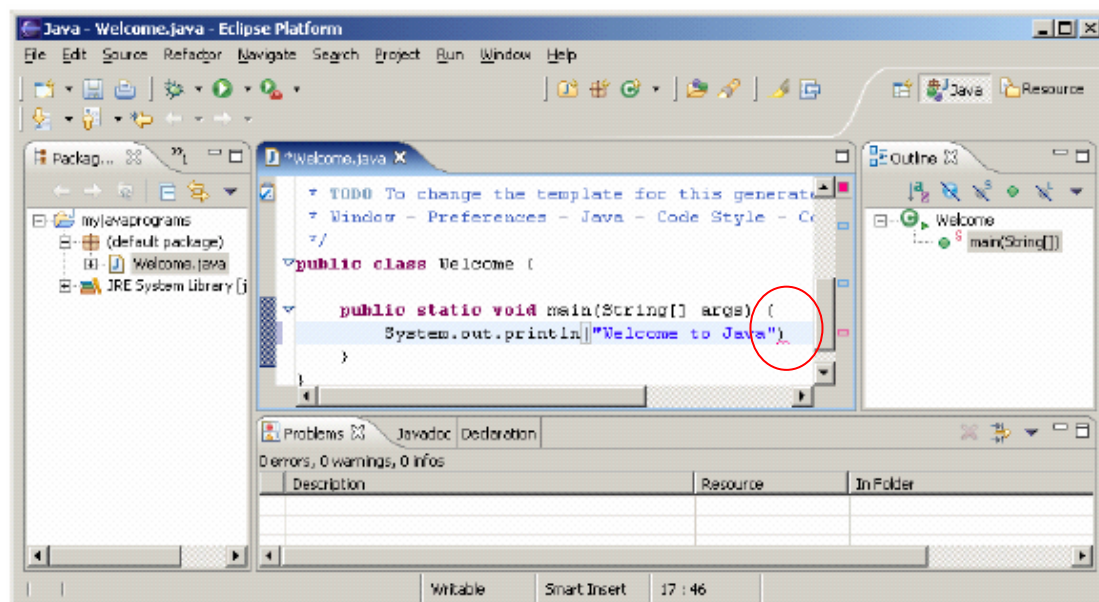


图 11

在运行程序前要保证程序是正确的并被编译通过了，换句话说就是程序中没有红色的波浪线了。

要运行程序 `Welcome.java`，右键点击 Package 视图中的该程序，在弹出的菜单中选择 `Run -> JavaApplication`，见图 12，程序运行的结果显示在控制台的窗格中，见图 13。

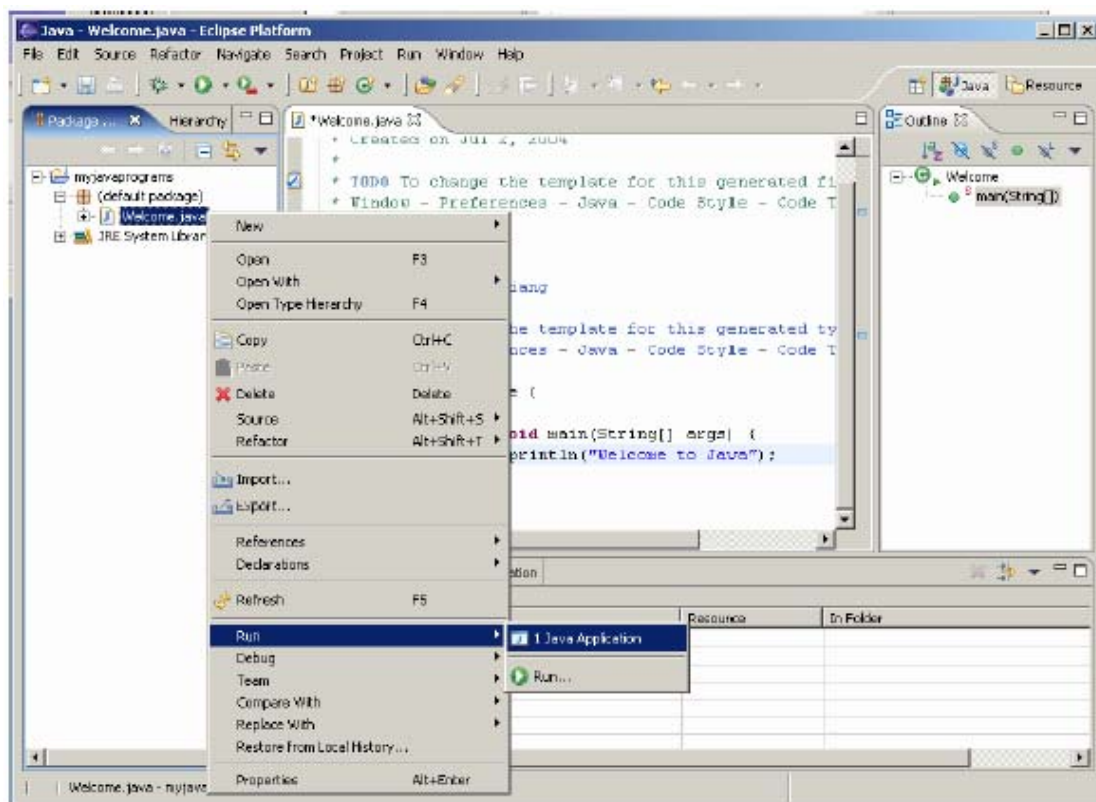


图 12

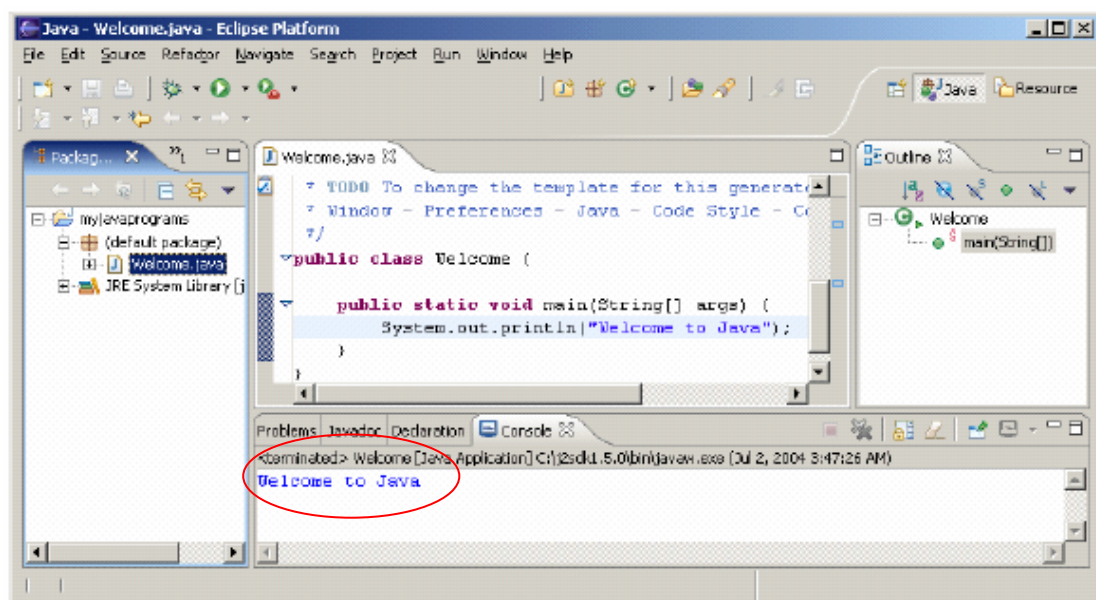


图 13

6 在命令行运行 Java Applications

除了使用工具软件来运行 Java 程序，还可以直接在操作系统上单独运行 Java 程序。下面是在 DOS 命令提示符下运行 Java 程序的步骤：

1. 点击 Windows 左下角的**开始 -> 程序 -> 附件 -> 命令提示符**打开 DOS 窗口。

2. 输入下面的命令来设置 Java 程序运行的环境

```
set path=%path%;c:\j2sdk1.5\bin
```

```
set classpath=.;%classpath%
```

3. 输入命令

```
cd c:\smith\myjavaprograms
```

到存放你所编写程序的目录下。

4. 输入命令

```
java Welcome
```

来运行程序。运行输出的界面如图 14 所示：

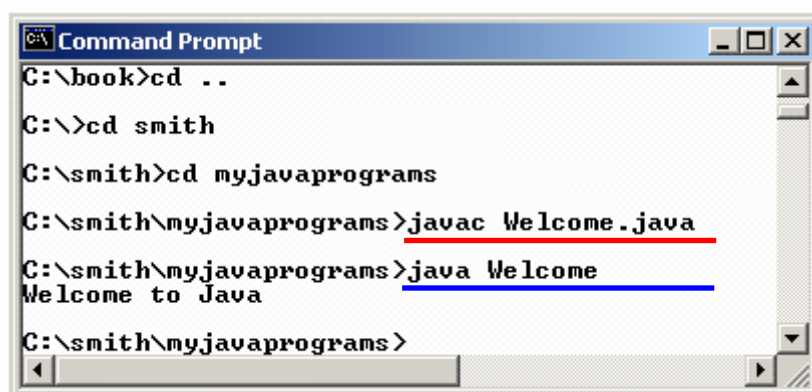


图 14

提示：你也可以使用 `javac Welcome.java` 命令在 DOS 提示符下编译程序 `Welcome.java`，如图 14 所示。

7 调试

Eclipse 还集成了调试工具。你可以不用离开集成环境就可以通过 Eclipse 调试器的帮助来找到错误。Eclipse 调试器提供了断点设置的功能使你可以一行一行的执行程序。在程序执行的同时，你可以看到变量存储了什么值，研究哪一个方法被调用了，并且知道程序将要

发生什么事件。

我们通过例 2.4（显示当前时间）来说明程序如何调试。在工作区 c:\smith 下创建一个新的类 ShowCurrentTime.java，程序内容如书 p61 例 2.4。

7.1 设置断点

你可以通过一行一行的执行程序来追踪它，但是对于调试大型程序来说这样很费时间。通常，我们知道程序中哪些部分运行正常，这些部分就不需要追踪，我们只需要追踪有可能出现问题的代码行，这种情况下，我们就可以使用断点。

断点是放置在源程序中告诉调试器到这一行暂停的标志。调试器依次运行程序直到遇到断点停止，所以你可以追踪在设置断点的那部分程序。使用断点，你可以很快的越过工作正常的程序从而找到引起问题的那部分程序。

有几种方式可以在一行上设置断点，最快的方法就是在你要设置断点的那一行的开始处双击，你会看到这行被选中，见图 15。你还可以通过菜单项 Run -> Toggle Line Breakpoint。要删除断点，双击该行的断点处即可。

在你调试程序时，你可以设置任意多个断点，在调试的过程中也可以随时删除断点。在你退出项目的时候 Eclipse 会在项目中保留你设置的断点。在你重新打开项目的时候断点仍然保存在项目中。

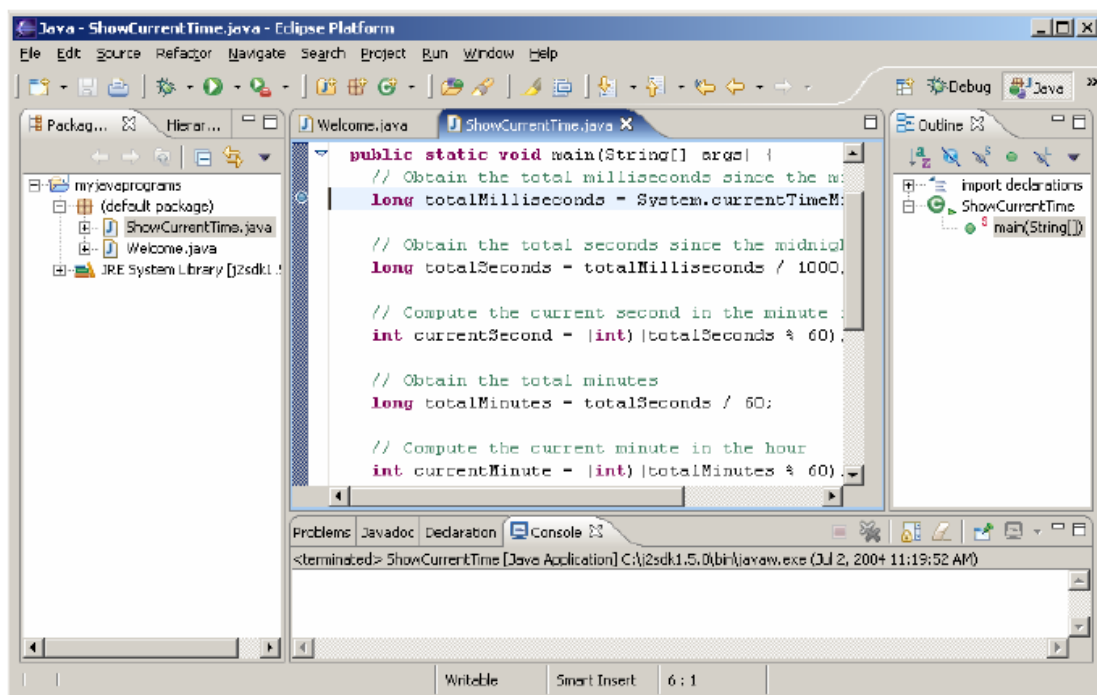


图 15

7.2 启动调试器

有几种方式可以启动调试器，一种简单方式如下所示：

1. 在代码编辑器中，在 main 方法的第一句设置断点。
2. 在项目面板右键点击 ShowCurrentTime.java 在弹出的菜单中选择 Debug ->Java Application 来启动调试器，你就会看到对话框 Confirm Perspective Switch，如图 16，点击按钮 Yes 进入调试透视图。



图 16

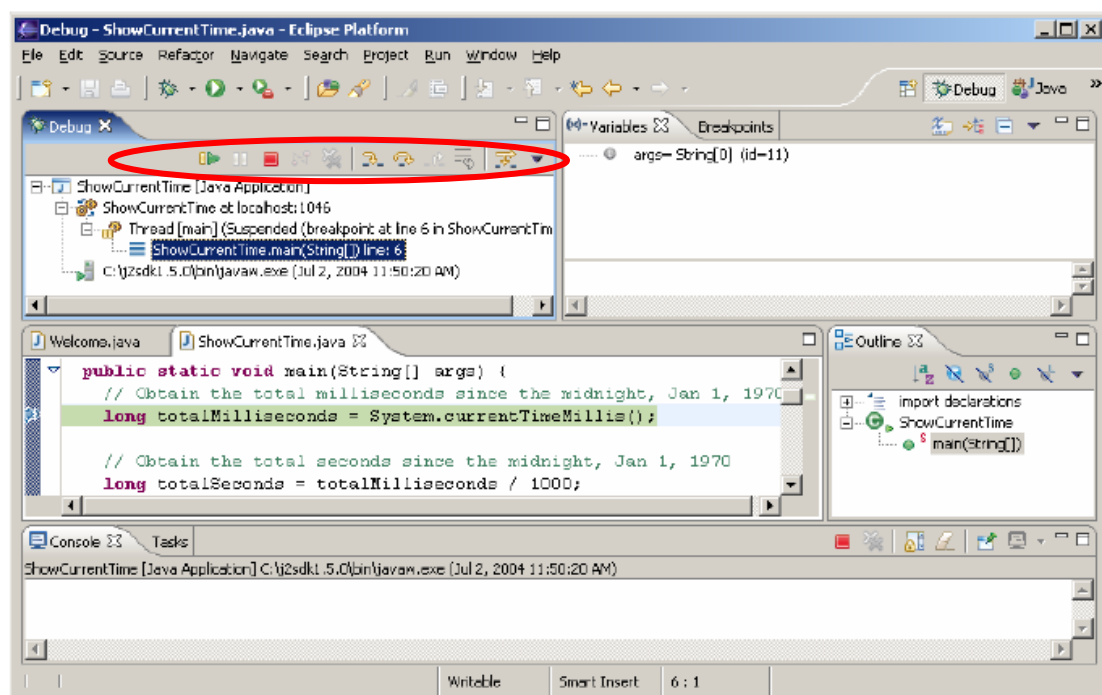


图 17

7.3 控制程序的执行

程序会停在 main 方法的第一行（整个程序的第 6 行），这一行，称为当前正在执行的

断点，用绿色背景标识和强调出来。正在执行的断点也指明了调试器要进行调试的下一行代码。

当程序在执行断点处停止时，你可以发表调试意见来控制程序的执行。你也可以预测和修改程序中变量的值。

当 Eclipse 处于调试状态下时，工具栏用于调试的按钮显示在调试窗口，见图 17 红色圈。工具栏的按钮也出现在主菜单 Run 中。以下是控制程序执行的命令：

Resume 继续执行被暂停的程序

Suspend 挂起，暂停一个程序的执行。

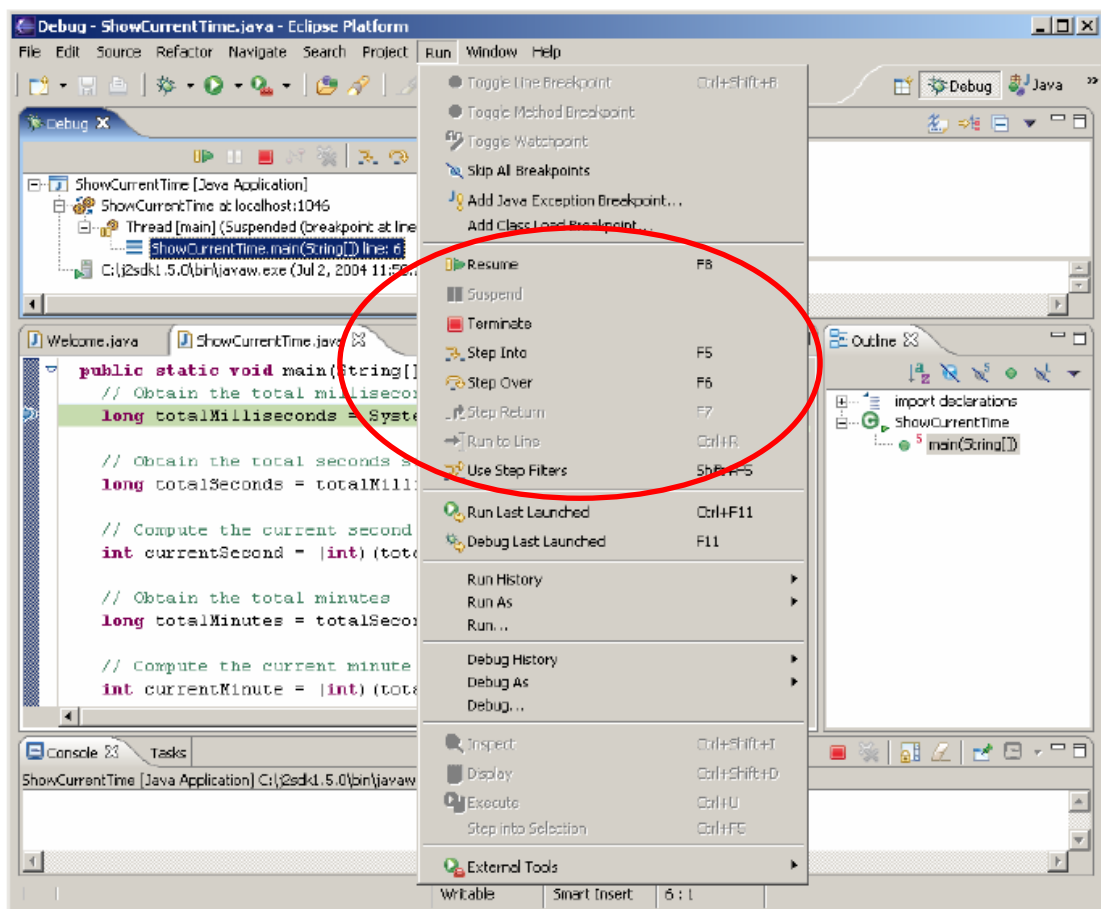
Terminate 终止，停止当前的调试活动。

Step Into 逐步执行，执行一条语句，或者步入一个方法。

Step Over 执行整体，执行一条语句，如果这条语句包含对一个方法的调用，那么不中断的执行完这个方法。

Step Return 返回，执行完当前方法中的所有语句后返回调用这个方法的主方法。

Run to Line 从开始处运行程序，到正在执行的断点暂停。



7.4 检查和修改变量

大多数功能强大的集成调试器都具备检查变量、数组元素、对象的值和方法调用过程中参数值的功能。如果你想使用一个新的变量值进行调试而不想重新启动程序，你甚至还可以直接在调试器中修改变量的值。

为了说明这个功能，选择 Run -> Step Over to execute one line，可以看到在变量面板上显示出了变量 totalMilliseconds 的值，如图 19。

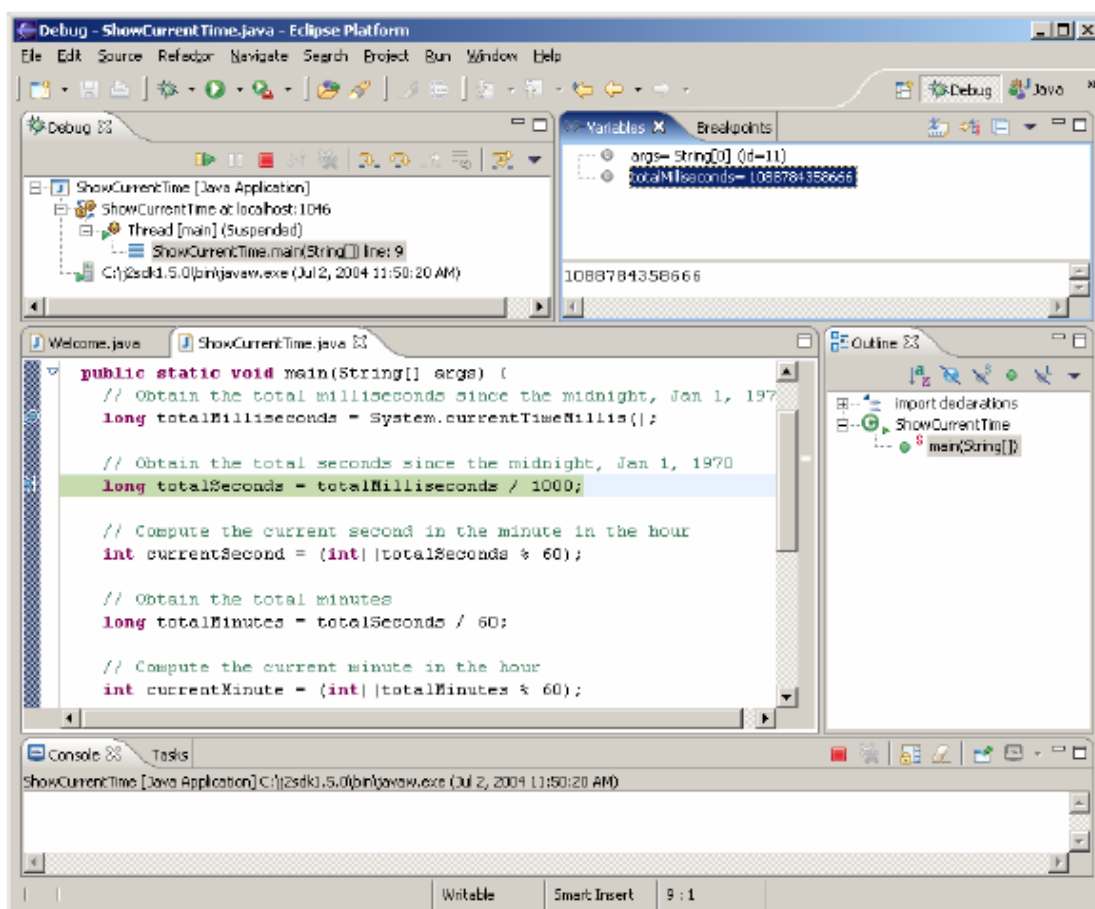


图 19

提示:

调试器是一个不可缺少的、功能强大的工具，它可以提高你的编程量。一开始你需要一些时间去熟悉它，但是你的努力会在将来的很长一段时间得到回报。

注:

调试器不仅仅是一个查找错误的重要工具，也是一个学习编程的很好的教学工具。

在调试后可以选择 java 透视图切换窗口，Window -> Open Perspective -> Java。

8 创建和测试 Java Applets

你可以用与创建 Java Application 同样的方法来创建 Java Applet，例如，14 章类 WelcomeApplet.java，如图 21。

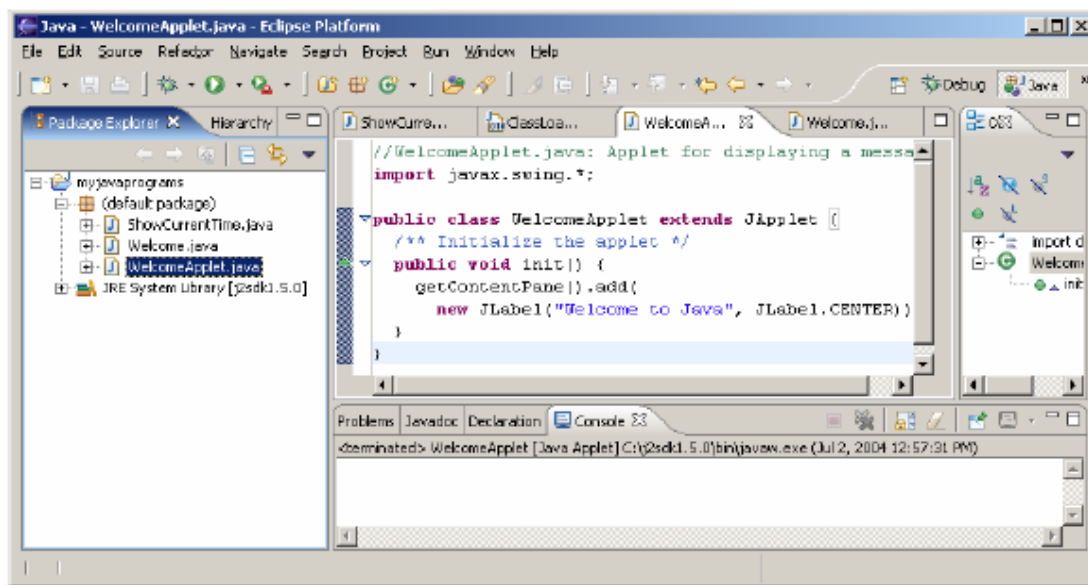


图 21

要运行一个 applet 程序，选择 Run -> Run as -> Java Applet，如图 22 所示，Eclipse 会自动为该程序创建一个 HTML 文件并且将 applet 包含进去，然后启动 appletviewer 工具来运行 applet，如图 23 所示：



图 23